
Contents

Purpose of Manual	1
Arrangement of Manual	1
Computer Characteristics and Communication Concepts	3
Computer Hardware	4
The “Brain” of the Computer	4
CPU Comparisons	5
Storing Information	7
Disks	7
Floppy Disks	7
Fixed Disks	8
Memory	8
How the CPU Uses Disks and Memory	9
What is a Byte?: How Memory and Disks are Measured	10
Ethernet Communication History	11
Communication Overview	12

Intentionally Blank

Purpose of Manual

The purpose of this manual is to provide information about Frameworkx Scoring Systems and the technologies utilized in this system. Information is provided in a nontechnical manner, as well as an advanced manner, allowing both the novice and experienced technicians to use the manual effectively. It is intended that this manual be informative and instructional, as well as provide quick solutions to any problem that may be encountered.

Arrangement of Manual

The *Frameworkx Scoring Systems Service Manual* is divided into individual sections by topic or component. To find a topic, turn to the appropriate tab. Each section includes its own Table of Contents to help the user find a topic within that section. A complete Table of Contents of all sections is provided at the beginning of this manual.

The *Frameworkx Scoring Systems Service Manual* is arranged as follows:

Section 1. *Introduction* explains the purpose and arrangement of this manual. It also provides information on computer terminology, computer operation, and network communication as it applies to Frameworkx Scoring Systems.

Section 2. *System Overview* provides block diagrams of the five configurations available in Frameworkx Scoring Systems, along with explanations of their components.

Section 3. *Scorers* provides an explanation of the circuit boards and boxes used in Primary and Secondary Consoles.

Section 4. *Center Management System* gives an explanation of the equipment that allows the Control Desk operator or other bowling center personnel to communicate with the lanes or bowler.

Section 5. *CCD Scanner* details the setup and adjustment of the CCD Scanner. Also described are the pinsetter modifications needed to allow the scanner to work properly and to connect the CMS System into the pinsetter.

Section 6. *Pinsensors* details the setup and operation of the Pinsensor PCB.

Section 7. *Pinsetter Interface* explains the functions and components of the Pinsetter Interface Box.

Section 8. *27" Ceronix Overhead* details the functions of the components within the 27" Ceronix Overhead and explains the adjustments necessary to properly maintain the unit.

Section 9. *BowlerTrack* explains the details of the BowlerTrack System and the adjustments needed to keep the system operational.

Section 10. *Instant Replay* explains the features of Instant Replay and the adjustments needed to keep the system operational.

Section 11. *Appendices* includes special maintenance procedures, system switch and jumper settings, repair center and support information, and a glossary of terms.

Computer Characteristics and Communication Concepts

A computer can help accomplish many complex tasks, but its processor actually performs only one function; it keeps track of whether a wire has voltage (or current) flowing through it. The turning on or off of this current is accomplished by the use of millions of switches and a concept referred to as *digital electronics*. Each switch manages a small amount of data, referred to as a *bit*, and can have only two values, one (1) or zero (0), which corresponds to the on and off operation of the switches. The term *bit* is derived from the expression binary digit. (Binary is a numbering system that is restricted to two values [the prefix “bi” means two]). Computers function by manipulating these numbers.

A bit, by itself, is not very useful because it can only represent two values, 0 and 1. When the bits are combined, however, they can represent almost anything. An example of this can be seen in the structure of the communication system known as Morse Code. This code uses only two symbols, dot and dash. When these symbols are strung together, they represent letters. When combined, these letters form words and sentences.

Computers combine the bits to represent more complex information or data. Typically, a computer will combine eight (8) bits together to form a unit called a *byte*. An 8-bit pattern can be arranged into 256 different combinations that can represent just about anything. For example, the eight bit pattern 01000001 represents the decimal number 65 or the letter “A” in a coding system called ASCII (American Standard Code for Information Interchange). (Other coding systems may use the same combination to represent characters or symbols as needed.)

The “Brain” of the Computer

At the core of every computer is a device known as the central processing unit, or CPU. This device is the “brain” of the computer which reads and executes program instructions, performs calculations, and makes decisions. The CPU is responsible for moving information from one part of the computer to another. For this reason, some people compare it to a central switching station or control tower that directs the flow of traffic throughout the computer system.

In personal computers, the CPU (also known as the microprocessor) is comprised of a single integrated circuit, or IC. An integrated circuit is a matrix of transistors and other electrical circuits embedded in a small slice of silicon. (Transistors are essentially microscopic electronic switches: tiny devices that can be turned on and off.) Like the dozens of other integrated circuits that inhabit a computer, from the outside, a CPU chip looks something like a square ceramic bug with little metal legs. These “legs” are designed to fasten the chip to a fiberglass circuit board inside the computer and carry electrical impulses in and out of the chip. Inside the ceramic case is the chip itself, a slice of silicon about the size of a fingernail. At first glance, it's hard to imagine how this tiny device can run an entire computer. But under a microscope, the slice of silicon reveals an electronic maze so complex it resembles an aerial photograph of a city, complete with hundreds of intersecting streets and hundreds of thousands of minuscule houses. Most of these “houses” are transistors with somewhere between one hundred thousand and three million of them on a single CPU chip. The type of CPU that a computer contains determines its processing power and how fast it can execute various instructions. Today, most CPUs can execute millions of instructions per second.

In Frameworkx Scoring Systems, the CPU, along with various support devices and tools for storing information, resides inside an area known as the *Lane Group Processor (LGP)*. Other support devices and tools for storing information will be described in further detail later in this chapter. For now, just think of the LGP as a container for the CPU.

People often categorize computers by the model of CPU chip they contain, saying phrases such as “I have a 486” or “My computer is a Pentium.” The CPU chips currently used in PCs include the 8088 at the low end, 80286, 80386, 80486, and at the high end, the Pentium (the generation after the 486 or 586). People usually omit the first two digits when referring to 80286, 80386, and 80486 chips. For example, a 386 computer is actually an IBM-type computer with an 80386 chip. One can tell that it's an IBM Computer or clone because they are the only computers that use 80386, 80486, or Pentium chips.

Within each class of CPU, speed is measured in terms of the cycle time at which the computer was designed to operate. All computers have built-in clocks that help regulate the flow of information from one part of the computer to another. Each pulse of this clock is known as a *cycle* and most CPUs can perform, at maximum, one operation per cycle. (New generation chips, such as the Pentium series, can perform two instructions per cycle.)

Every CPU is designed to work with a clock that “ticks” at a particular rate. Originally, chips were designed to run at 16 megaHertz (MHz) or 33 MHz meaning 16 or 33 million cycles per second. However, new chips can exceed 700 MHz. This measurement is only relative when comparing processors of the same class. For example, a 400 MHz Pentium is faster than a 100 MHz Pentium, but a 100 MHz Pentium CPU runs much faster than a 100 MHz 486 CPU. The reason for this has to do with the internal design of the chips. (Refer to *CPU Comparisons* for more information.)

CPU Comparisons

When comparing CPU performance, it is important to understand the terminology used to describe their operation. The following terms and definitions are most often used when talking about microprocessors.

Bus - The physical path information takes when transferred from one device to another. A bus may be nothing more than the metal circuit trace on a circuit board. The size or width of the bus refers to the number of paths available to move information or data. Currently the number of paths available can range from 8 to 64.

Cache - An amount of Ram set aside to hold data that is expected to be processed again. By storing the data in this special area access to it becomes much faster which allows for faster processing.

Clock Speed - A measure of how fast the CPU is operating and how fast the CPU can process information. The clock speed is measured in MHz (1 MHz = 1 million cycles per second).

Coprocessor - A circuit or circuit chip that allows complex mathematical calculations without time consuming binary conversions. This circuit is useful for any application that requires intense calculations such as spreadsheets or engineering tools like CAD.

External Data Bus - The number of paths available to the CPU for moving information from the CPU to outside devices. Currently, most CPUs have either a 16, 32, or 64 bit external data bus.

Internal Data Bus (size) - The number of internal paths available in the CPU to move information. This value can be 8, 16, 32 or 64 bits.

Memory Address (size) - The number of bits available to identify or reference individual memory locations or areas. As explained earlier, an 8 bit system or pattern can represent 256 different combinations and therefore, can identify only 256 different memory locations. Currently, most CPUs have a 24 or 32 bit memory address size. This allows them to identify either 16 M (16,000,000) or 4 G (4,000,000,000) different locations.

Register - Areas inside a CPU where information is temporarily stored as it moves from one place to another along the internal data bus.

Refer to the following table for comparison of microprocessors.

	8086	80386SX	80386DX	80486SX	80486DX	Pentium, Pentium II, Pentium III
Internal Bus Size	16 Bit	16 Bit	32 Bit	32 Bit	32 Bit	64 Bit
External Bus Size	16 Bit	16 Bit	32 Bit	32 Bit	32 Bit	64 Bit
Address Size	20 Bit (1 MB)	24 Bit (16 M)	32 Bit (4 G)	32 Bit (4 G)	32 Bit (4 G)	32 Bit (4 G)
Speed	4.77- 10 MHz	16-25 MHz	16-33 MHz	16-25 MHz	25-33 MHz	60-850+ MHz
Built-in Math Co-processor	No	No	No	No	Yes	Yes
Built-in Cache	No	No	No	Yes (8 K)	Yes (8 K)	Yes (8 K-512 K)

CPU Comparison Chart

As shown in the *CPU Comparison Chart*, the 80086 microprocessor in the original PC was capable of operating at a speed of 4.77 MHz per second. This meant that 4,700,000 instructions or parts of an instruction were executed each second. In addition, 80086 had an address size of 20 bits. This meant that it could access 1 MB of memory (1 million memory locations) and could only move information along a 16 bit bus. This made the microprocessor very slow and inefficient when compared to the microprocessors of today. In contrast, the 386 and later CPUs can access 4 G of memory (4 billion memory locations) and move twice as much data along a 32 bit bus (4x in the 64 bit Pentium).

For Frameworkx Scoring Systems, the CPU in each Lane Group Processor must be at least a 386SX in order to obtain the speed and processing capability required by the system. To increase efficiency, most LGPs have a 486 CPU.

Storing Information

Although the CPU is terrific at manipulating data and following instructions, it has almost no capacity for storing information. In order to function, your computer needs a place to store both programs (instructions that tell the CPU what to do) and data. You need, in other words, the electronic equivalent of a closet or filing cabinet. Two methods typically used to store information are disks (fixed or floppy), and memory (RAM [Random Access Memory] and ROM [Read Only Memory]).

Disks

In most computers, primary storage places are called disks (flat, circular wafers that resemble undersized phonograph records). Like phonograph records or compact disks, they store information that can be “played” by devices specifically designed for that purpose. The device that “plays” computer disks is known as a *disk drive*. In several respects, it is the equivalent of a turntable or CD player. Like turntables or CD players, disk drives have components designed to access the information on a specific area of the disk. These parts are called *read/write heads* and are equivalent to the phonograph needle on a turntable or laser in a CD player. Like turntables, disk drives turn around, thereby spinning the disk so that different parts of its surface pass underneath the read/write heads (just as records spin underneath the needle). Most disk drives have at least two read/write heads, one for each side.

Unlike record players or CD players, however, disk drives can record new information on disks as well as play existing information. (In this sense, they are more like cassette tapes than records.) In computer terminology, the process of playing a disk is called *reading* and the process of recording onto a disk is called *writing*. Hence the term read/write.

Computer disks come in two basic types: floppy and fixed.

Floppy Disks

Floppy disks generally hold less information and are slower than hard disks. They can also be removed from their disk drives. In other words, you can “play” different floppy disks in the same drive by removing one and inserting another. The word *floppy* refers to the disk itself which is a very thin, round piece of plastic on which information is magnetically recorded (much as music is recorded on the surface of plastic cassette tapes). This floppy disk is enclosed inside a sturdy, unfloppy plastic jacket to protect it from dust, scratches, liquid, and the oils on your fingers, all of which can erase or scramble the information recorded on the disk’s surface. The disks used in personal computers are usually 3-1/2 or 5-1/4 inches in diameter, although the 3-1/2” type is the standard.

Fixed Disks

Fixed disks, also referred to as hard disks, are designed to hold more information and spin faster than floppy disks (about twenty times faster). They are permanently enclosed within their disk drives. You can think of fixed disks as records that are encased inside their record players so that the record and the record player function as a single unit. Contrary to popular belief, hard disks are not always larger than floppies. They're simply capable of packing information more tightly and therefore, can store more data in the same amount of space. Some hard disk drives contain multiple disks often called *platters*, which are stacked vertically inside the drive. Typically, each disk has its own pair of read/write heads. Since you never remove the actual disks, hard disk drives do not contain doors or slots, as do their floppy counterparts. In Frameworkx Scoring Systems, a hard drive stores the program for the lane pair.

Memory

There are two types of memory used in personal computers, read-only memory (ROM) and random-access memory (RAM). The ROM chips have software (program instructions) permanently etched into their circuitry and are often referred to as *firmware* because it is halfway between hardware and software. RAM chips are used to store information referred to as *data*, that is currently being worked with.

Unfortunately the terminology is misleading because both RAM and ROM allow random access. To properly distinguish RAM from ROM, RAM should be described as read/write memory. Information can be read (retrieved) from RAM and written (recorded) to it as well. In contrast, ROM memory, in its most basic form, has instructions that are permanently “frozen” into its circuitry. The feature that sets RAM apart from ROM is its changeability, the fact that you can alter its contents at will.

NOTE: ROM is also available in special variations that allow the contents to be reprogrammed or altered. Frameworkx Scoring Systems uses two varieties of ROM called EPROM (erasable programmable read-only memory) and Flash ROM. EPROMs can be reprogrammed with a special programming device. Contents of Flash ROM can be altered directly by the computer.

The other difference between RAM and ROM is how long their memories last. RAM is short term memory. It forgets everything it knows as soon as the computer is turned off. ROM is long term memory. It remembers everything even if the power is removed.

In personal computers, ROM is generally used to store some part of the operating system. In IBM-type computers, only a small part of the operating system is stored in ROM, just enough to get the hardware up and running and to tell the CPU how to locate and load the rest of the operating system from disk.

How the CPU Uses Disks and Memory

Given what you've learned so far, you might assume that when you run a program, the CPU retrieves instructions from the disk or ROM chips and executes them, returning to the disk drive or ROM every time it finishes a single step. If this were actually the way computers worked, they would be so slow they would be unusable.

On their own, most personal computer CPUs are capable of executing between one million and one hundred million instructions per second. But because of its slow mechanical moving parts, disk drives cannot deliver program instructions anywhere near that fast. Reading an instruction from the disk involves both rotating the disk so that the proper section is below one of the read/write heads and then moving the head closer to or farther from the center of the disk until it is positioned directly above the spot where the instruction is recorded. Even on a hard disk drive, this process generally takes between 9 ms and 15 ms (ms = milliseconds = 1 thousandth of a second). Additionally, the ROM chips are extremely slow when compared to the CPU and cannot deliver program instructions quickly enough.

If the CPU can execute several million instructions per second and the disk drive or ROM can only deliver one hundred thousand instructions per second, you have the equivalent of an assembly line in which one person on the line (the CPU) is moving somewhere between 10 and 100 times faster than the previous one (the disk drive). If this were really the way your computer worked, the speed of the CPU would be wasted while it waited for delivery of the next instruction.

For the computer to function efficiently, it needs a place to keep information that is capable of keeping pace with the CPU. This extra piece is the RAM (random access memory). Physically, RAM consists of a set of separate integrated circuits (each of which looks something like a small CPU chip) which are often mounted on fiberglass board (reference *SIMM*). In practice, however, the CPU treats it as a single, contiguous storage area. The CPU uses the memory like a set of mailboxes inside a post office. Each mailbox holds a single piece of information and the entire collection of boxes is numbered sequentially. (In computer jargon, the data in the mailbox is called a *byte* and their numbers are known as *memory address*.)

Like the CPU chip, memory chips store and transmit information electronically. Sending instruction from memory to the CPU is therefore a simple matter of transmitting electrical impulses.

Because the CPU can move information in and out of memory so quickly, it uses memory as an “electronic desktop”. The electronic desktop stores whatever the CPU is working on or plans to work with shortly. When a particular program is used in a computer, for example, the first thing the CPU does is find the program on the hard disk or ROM chip and copies it into RAM memory. This process is known as “loading a program”. This gets the comparatively slow process of reading instructions from disk over with at the start. Once the entire program has been loaded, the CPU can quickly read instructions from memory as needed.

Although you can compare placing a program in memory to moving it from a file cabinet (the disk or ROM) to your desktop (RAM), there is one important difference. Placing something on your desktop entails removing it from its usual storage place in the file cabinet. In contrast, when you place a program in memory, you do not remove it from anywhere. Instead, you make the equivalent of a photocopy of the program, and place the copy in memory. The original copy of the program stays on the disk or in the ROM chip, ready to load again whenever you need it.

The first program that is loaded into memory in every work session is the operating system. In fact, just about the first thing a CPU does when you turn on a computer is locates and loads the operating system program. This program then remains in memory until you turn off the computer. When you load programs for specific applications, they always share the electronic desktop with the operating system. In fact, application programs need to have an operating system in order to function.

What is a Byte?: How Memory and Disks are Measured

As mentioned, the term *byte* means the amount of space required to represent a single character, a letter, a number, or even a space. (In the mailboxes analogy, it's a single mailbox.) This term is used regardless of whether you're talking about space in memory, on a disk, or on any other storage medium. Since many, many bytes are often required to accommodate an entire database, program or instruction, computer language includes terms to describe these larger units of measurement.

Kilobyte - A kilobyte (KB or kbyte) is 1,024 bytes. To a computer's way of thinking, 1,024 is a nice round number. Computers “think” in units of two and 1,024 is 2 to the 10th power. Most people simply think of a kilobyte as “around 1,000” bytes. So when someone tells you that they have a computer with 640K of RAM, they mean a computer that has 640 kilobytes or approximately 640,000 characters worth of random-access memory. A floppy disk that holds 360K of data can accommodate a little more than 360,000 characters.

Megabyte - The term megabyte (MB or Mbyte) is 1,024 kbytes or approximately one million bytes. Most 3-1/2" floppy disks hold 1.44 MB of data.

Gigabyte - The term gigabyte (GB) means a kilobyte to the third power (1,024 times 1,024 times 1,024), or approximately one billion bytes.

A few years ago most computers had less than a megabyte of RAM. Each LGP's Motherboard in Framework Scoring Systems comes standard with 4 MB of RAM which can be increased up to 128 MB. Hard disks typically hold between 500 MB to several gigabytes. The Framework LGP contains a 810 MB or higher drive.

Ethernet Communication History

The Ethernet System was originally developed in the 1970s at Xerox Corporation's Palo Alto Research Center to link its workstations to laser printers. During its first years, Ethernet was proprietary to Xerox. It was a technology without a purpose in a world in which the PC had not yet been invented.

In September of 1980, however, Xerox joined with minicomputer maker Digital Equipment Corporation and semiconductor manufacturer Intel Corporation to publish the first Ethernet specification. The original specification was followed in November 1982 by a revision that became today's widely used standard.

This original specification is not what most people call Ethernet, however. In January 1985, the Institute of Electrical and Electronic Engineers (IEEE) published a networking system derived from the Xerox Ethernet, but not identical to it. The result was a new standard, the IEEE 802.3 specification. The original Ethernet and IEEE 802.3 share many characteristics. Physically, they use the same wiring and connection schemes, but each uses a different method for arranging the information being sent. Consequently, although you can plug host adapters for true Ethernet and IEEE 802.3 together in the same cabling system, the two standards will not be able to talk to one another. Some PC host adapters, however, are able to speak both languages and can exchange packets with either standard.

Communication Overview

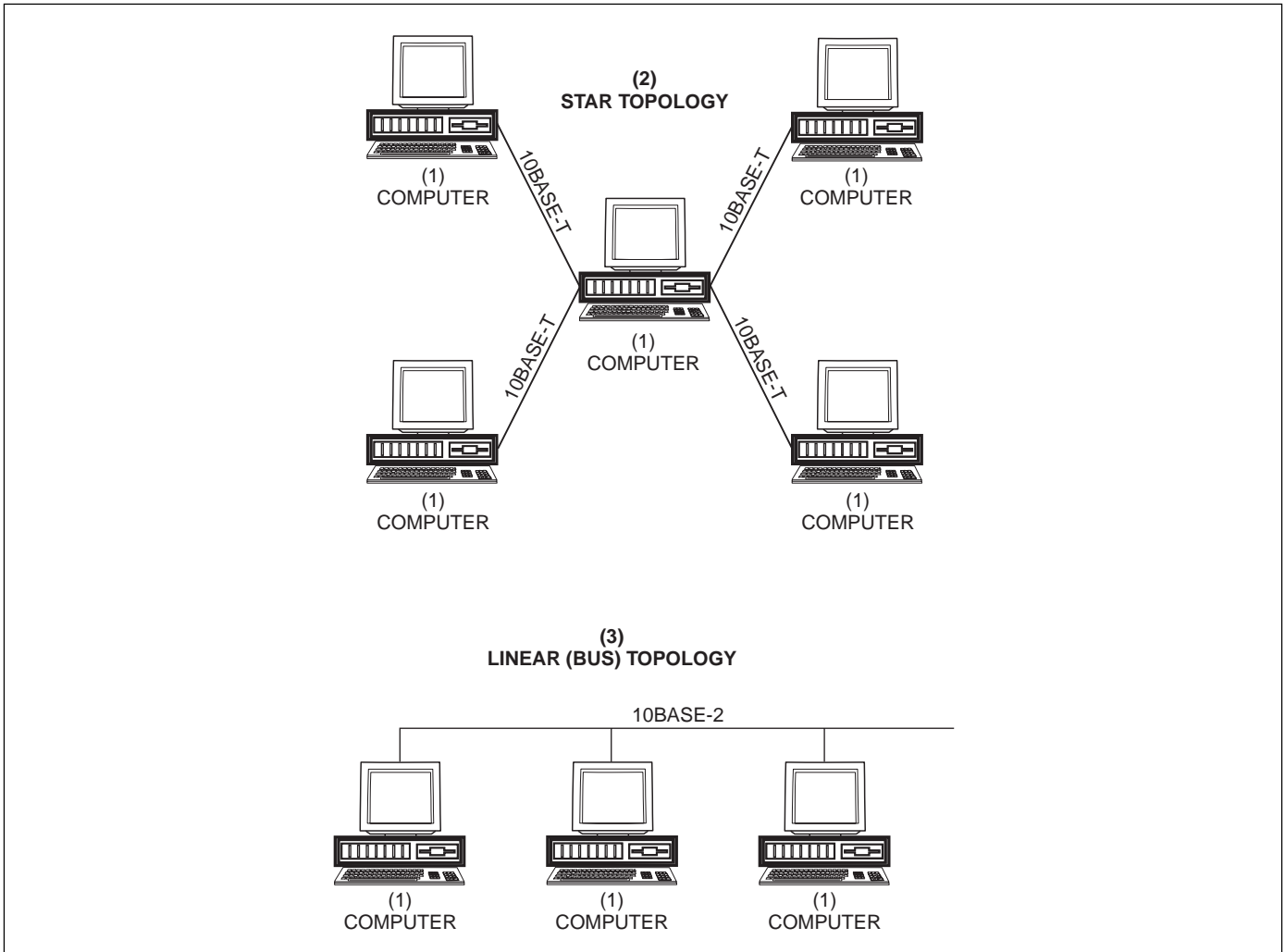
The basis of Ethernet is a clever scheme for controlling how individual computers access the central communication lines within a larger system. The protocol, formally described as *Carrier Sensing, Multiple Access with Collision Detection* (CSMA-CD), is often described as being like a party line. It's not. It's much more like polite conversation. All PCs on the network patiently listen to everything that's going on across the network lines. Only when there is a pause in the conversation will a new PC begin to speak. (PCs do this by sensing a signal called a *carrier*, hence, *carrier sensing*. When a carrier is not present, any PC can begin talking - *multiple access*.) If two or more PCs start to talk at the same time, the information being sent will interfere with each other and become garbled. The computers will sense this and all will become quiet. This is called *collision detection*. Each PC will then wait for a random (and different) period of time before attempting to speak again. One will be lucky and win access to the network. The other unlucky PCs will hear the first PC talking and wait for another pause.

Access to the network line is not guaranteed by the Ethernet protocol. The laws of probability guide the system and dictate when every device that desires access will eventually get it. Consequently, Ethernet is described as a probabilistic access system. As a practical matter, when few devices attempt to use the Ethernet System, delays are minimal because all of them trying to talk at one time is unlikely. As demand approaches the capacity of the system, however, the efficiency of probability-based protocol plummets. Therefore, the size limit of an Ethernet System is not set by the number of PCs, but by the amount of traffic. The more information packets PCs send, the more difficult it will be for other PCs to gain access.

The Ethernet protocol has many physical characteristics. These include how it is wired, the type of cable used, and the speed. The IEEE Specification defines several of characteristics and assigns a code name to each. Today's most popular Ethernet implementations operate at a raw speed of 10 MHz or 100 MHz. That is, the clock frequency of the signals on the Ethernet (or IEEE 802.3) wire is 10 MHz or 100 MHz.

Framework Scoring Systems use two of the most popular types of Ethernet cabling; 10Base-2 and 10Base-T. Although confusing at first look, you can remember the names as codes. The first number indicates the operating speed of the system in megahertz. The central word "base" indicates that Ethernet protocol is the basis of the system. The final character designates the wire used for the system. The final digit, when numerical, refers to the approximate distance in hundreds of meters that the cable length can stretch, but as a practical matter also specifies the type of cable used. Coincidentally, the number also describes the diameter of the cable. For the 10 MHz Ethernet System described above (10Base-2 and 10Base-T), the "2" refers to a thinner coaxial cable about two-tenths inch in diameter that should not be extended beyond 185 meters (607 feet). The "T" indicates twisted-pair wiring. This cable type should not extend beyond 100 meters (328 feet).

Another difference, besides cable type, that separates these Ethernet schemes is in their topology. A topology describes how the devices are connected together and can be thought of as simply a map of the network. The 10Base-2 is wired in a configuration known as a linear or bus topology. 10Base-T is wired in a star configuration. Refer to the figure titled *Star and Linear Topology*.



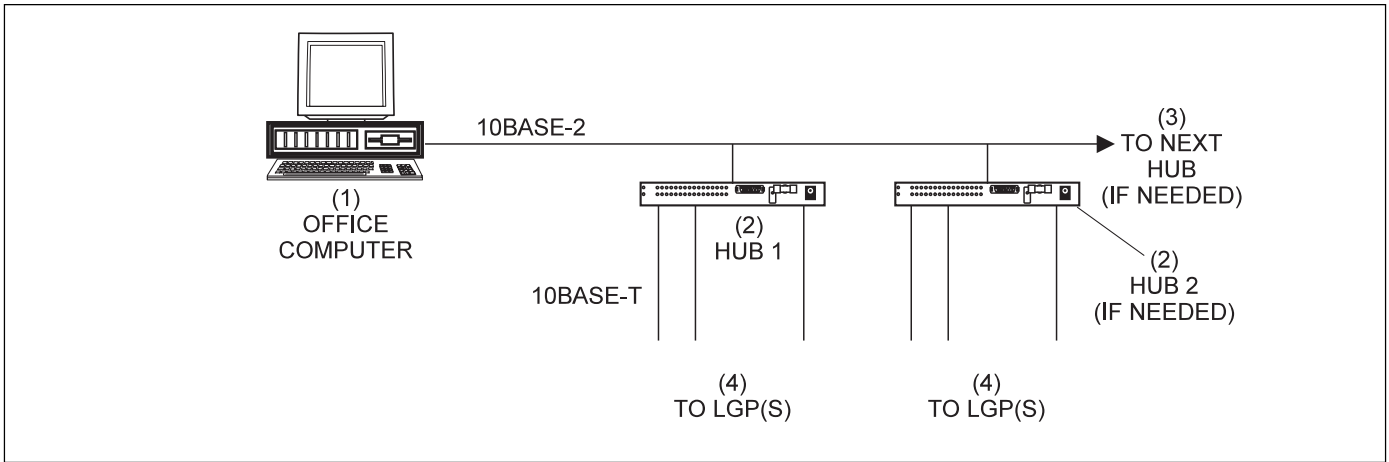
Star and Linear Topology

(1) COMPUTER

(2) STAR TOPOLOGY

(3) LINEAR (BUS) TOPOLOGY

Both systems operate at the same 10 MHz speed using the same Ethernet protocol, so a single network can tie together both technologies without the need for such complications as protocol converters. In typical complex installations, such as Frameworkx Scoring Systems, thin coaxial cable 10Base-2) links systems together that are far apart, while each system is tied together locally with a 10Base-T hub. The figure titled *Ethernet Cabling for Frameworkx* shows how the cables are used in Frameworkx Scoring Systems.



Ethernet Cabling for Frameworkx

(1) OFFICE COMPUTER
 (4) TO LGP(S)

(2) HUB

(3) TO NEXT HUB